

StreamGate Public API 介面文件

StreamGate 1.9.7 · 客戶交付文件

本文件說明第三方系統接入 StreamGate REST API 的方式，包含授權、驗證、權限範圍、資源介面、呼叫範例、錯誤處理、安全與版本相容資訊。

項目	內容
適用版本	StreamGate 1.8.2、1.9.1、1.9.2、1.9.3、1.9.4、1.9.6、1.9.7
API 版本	v1
文件版本	1.2
更新日期	2026-10-13
基礎位址範例	http://streamgate.example:18080
介面前綴	/api/public/v1
資料格式	application/json; charset=utf-8
驗證方式	Authorization: Bearer <API_TOKEN>
時間格式	ISO 8601；伺服器時間欄位使用 UTC

[streamgate.example](#) 是文件中的預留位址。實際接入時應替換為 StreamGate 伺服器的網域名稱或 IP，並保留實際設定的 Web 連接埠。

1. 文件範圍

Public API 僅供 Professional / Enterprise 部署使用，可讀取裝置、通道、執行狀態、播放位址、錄影索引與告警狀態，並提供裝置、通道、錄影、GB28181、WebHook 與 MQTT 管理介面。

接入方式	方向與用途	驗證方式
REST API	客戶系統主動請求網關，讀取或修改資源	Bearer Token 與所需 scope
WebHook	網關主動向客戶 HTTP 接收端 POST 事件	接收端校驗 HMAC 簽章
MQTT	網關向 broker/topic 發佈事件，客戶訂閱	broker 憑據、主題權限與設定的 QoS

三者不是同一介面。API 中的 [/integrations/webhooks](#)、[/integrations/mqtt](#) 管理訂閱與投遞設定，不是事件接收位址；API Token 也不替代 WebHook 簽章密鑰或 MQTT 憑據。

Professional / Enterprise 執行套件包含 [streamgate-public-api-v1.openapi.yaml](#)。請以該 OpenAPI 文件作為請求與回應結構的機器可讀權威合約；本文件用於說明接入流程、行為與運維邊界。

2. 版本相容

能力	1.8.2	1.9.1	所需 scope
查詢通道	支援	支援	channels:read
查詢錄影索引	支援	支援	recording:read
查詢告警	支援	支援	alarms:read
確認或清除告警	支援	支援	alarms:write
查詢裝置身分	不支援	支援	devices:read
查詢通道執行狀態	不支援	支援	status:read
取得播放位址	不支援	支援	playback:read
裝置與 ONVIF 管理	不支援	Professional	devices:write
通道設定與 PTZ	不支援	Professional	channels:write、ptz:write
錄影任務與策略	不支援	Professional	recording:read、recording:write
GB28181 管理	不支援	Professional	gb28181:read、gb28181:write
WebHook、MQTT 與投遞記錄	不支援	Professional	integrations:read、integrations:write

第三方程式應忽略回應中無法識別的欄位，讓後續相容版本能增加選用資訊。對 1.8.2 與 1.9.1 混合部署，應先探測所需端點；1.8.2 回傳 404 時按版本能力降級。

3. 授權、Token 與權限範圍

目前 1.9.7 的 Public REST API、WebHook、MQTT 均不向 Free / Standard 開放；有效 Professional 試用、Professional 或 Enterprise 軟體授權可使用，仍須滿足 [openapi](#) 權益與執行套件條件。評估影片不加浮水印；試用僅用於授權允許的非生產評估，期限、容量及功能範圍依簽發授權為準。REST 呼叫及透過 REST 管理集成還須有效 API Token 與相應 scopes；WebHook 簽章校驗、MQTT 憑據與主題權限另行滿足。Analysis 授權不替代軟體權益，控制台內部介面不屬於 Public API。歷史 1.8.2 介面仍要求已啟用 OpenAPI 的 Professional / Enterprise 授權，版本支援不代表自動授予介面權限。

以管理員登入後，在 [系統設定](#) > [開放介面](#) > [API Token](#) 建立 Token。填寫第三方系統名稱，按最小權限原則選擇 scope，並立即儲存只顯示一次的完整 Token。後續只會顯示前綴，無法還原原文；遺失時應撤銷舊 Token 並重新建立。

Scope	用途
channels:read	讀取通道清單與通道設定。
recording:read	讀取錄影索引、任務與策略。
alarms:read	讀取告警狀態。
alarms:write	確認或清除告警。
devices:read	讀取裝置身分與已脫敏的裝置設定。
status:read	讀取通道分發觀測狀態。
playback:read	讀取通道播放位址。

Scope	用途
playback:warm	短時保持通道來源就緒，與讀取播放位址分開授權。
devices:write	建立、更新、刪除、批次操作與探索裝置。
channels:write	建立、更新、刪除、批次操作與切換通道按需模式。
ptz:write	執行 PTZ 控制，與通道設定權限分離。
recording:write	管理錄影任務與策略。
gb28181:read	讀取 GB28181 平台、目錄發布與上聯。
gb28181:write	管理 GB28181 平台、目錄發布與上聯。
integrations:read	讀取 WebHook、MQTT、事件類型與投遞記錄。
integrations:write	管理與測試 WebHook/MQTT，清除投遞記錄。
exports:read	讀取已脫敏的 StreamGate 執行事件。
*	完整 Public API 權限，只應用於受控的可信整合。

請在請求標頭中傳送 Token：

```
Authorization: Bearer <API_TOKEN>
Accept: application/json
Content-Type: application/json
```

GET 請求可以省略 **Content-Type**。禁止把 Token 放入 URL、查詢參數、瀏覽器程式碼、截圖或一般應用程式記錄。

4. 快速驗證

```
curl --request GET \
  "http://streamgate.example:18080/api/public/v1/channels" \
  --header "Authorization: Bearer <API_TOKEN>" \
  --header "Accept: application/json"

$baseUrl = "http://streamgate.example:18080"
$headers = @{ Authorization = "Bearer <API_TOKEN>" }

Invoke-RestMethod `
  -Uri "$baseUrl/api/public/v1/channels" `
  -Headers $headers `
  -Method Get
```

預期狀態碼為 200，回應包含 **channels** 陣列。空陣列表示目前沒有已設定通道，不代表 API 異常。

5. 基礎唯讀資源

通道

GET **/api/public/v1/channels** 需要 **channels:read**。回傳已設定的通道，不包含裝置憑據或來源 URL。

欄位	類型	說明
channels[].id	string	穩定的邏輯通道識別碼。
channels[].endpointId	string	關聯接入端點，可選。
channels[].deviceId	string	關聯裝置，可選。

欄位	類型	說明
channels[].name	string	顯示名稱。
channels[].enabled	boolean	是否啟用。
channels[].inputType	string	輸入協定類型。
channels[].outputStreamId	string	分發串流識別碼。

錄影

GET /api/public/v1/recordings 需要 recording:read。查詢參數以 OpenAPI 為準，包含通道、時間範圍、分頁與排序。所有時間戳均按 UTC ISO 8601 處理。錄影路徑屬於伺服器部署資訊，應按敏感資料管理。

告警

GET /api/public/v1/alarms 需要 alarms:read，可按狀態、通道、嚴重度、時間範圍與分頁篩選。狀態變更使用回傳的 dedupeKey：

- POST /api/public/v1/alarms/{dedupeKey}/ack 需要 alarms:write，用於確認告警。
- POST /api/public/v1/alarms/{dedupeKey}/clear 需要 alarms:write，用於清除告警。

寫入逾時後，先重新查詢告警狀態，再決定是否重試。

裝置與執行狀態

GET /api/public/v1/devices 需要 devices:read 及 StreamGate 1.9.1 以上版本。回傳 id、name、vendor、protocol、enabled 等身分欄位，不回傳憑據與私有來源 URL。

GET /api/public/v1/status 需要 status:read。其結果是 StreamGate 目前觀測到的分發狀態，不會主動探測來源裝置，也不等同於裝置在線狀態。已知值為 unknown、starting、active、draining、idle、error；客戶端應允許後續相容版本增加狀態值。

6. 播放位址

GET /api/public/v1/channels/{id}/playback-urls 需要 playback:read。選用查詢參數 quality 可為 auto、high 或 fluent；未提供時採用已儲存的 viewing.defaultQuality，明確的值覆寫此預設值。未設定觀看選項的舊通道保留原單路行為。使用 auto 時可附帶 1 至 8192 的 viewportWidth 與 viewportHeight；傳入寬度不大於 640 或高度不大於 360 時選擇可用子碼流，較大視窗或未提供尺寸時優先主碼流。子碼流不可用時回退主碼流。

在「系統設定 > 媒體與播放」的「控制台播放碼流」設定 media.defaultViewingQuality: auto/high/fluent 分別為自動、主碼流、子碼流，統一用於控制台預覽、詳情及標準/專業大螢幕。自動依視窗與呈現方式選流，沒有可用子碼流時回退主碼流。通道介面已移除預設播放選擇；舊 viewing.defaultQuality 僅相容未傳 quality 的 Public API 播放位址請求，明確的 quality=high/fluent/auto 仍覆寫它。系統控制台選擇不改變 Public API、錄影選源或 GB28181 供流。

呼叫此介面不會啟動來源拉流，也不會簽發播放憑據。

欄位	說明
id、enabled、outputStreamId	邏輯通道身分與目前設定。
httpFlv	HTTP-FLV 位址；協定停用時為空。

欄位	說明
webSocketFlv	WebSocket-FLV 位址；協定停用時為空。
hls	HLS 位址；協定停用時為空。
webRtc	WebRTC 播放頁面位址；協定停用時為空。
whep	WHEP 端點位址；協定停用時為空。
requestedQuality、actualQuality	請求與實際選擇的畫質。
qualitySource、qualityReason、qualityFallback	畫質來源、原因與回退資訊。
fluentAvailable	是否具備子碼流輸出。
playbackTokenRequired	是否仍需要獨立播放授權。

API Token 只用於 Public API，不能取代 Web Console 工作階段或播放憑據。playbackTokenRequired 為 true 時，播放器仍須按部署的播放驗證流程取得有效憑據。

多視窗介面應在宮格狀態請求 ?quality=fluent 使用子碼流，放大時停止原播放器並重新請求 ?quality=high 使用主碼流，縮回宮格時再切回 fluent。每次均以 actualQuality 為準；未設定子碼流時，服務端會回退主碼流並透過 qualityFallback 說明原因。

high 對應通道主碼流，fluent 對應子碼流；既有 quality 參數、high/fluent 值、回應欄位與播放 URL key 均保留相容。選擇碼流不臨時建立轉碼策略。通道主碼流使用所選輸入，也可來自裝置子碼流，不等同於裝置實體主碼流；子碼流可來自裝置或由同一原始輸入獨立編碼，不是再次轉碼主碼流輸出。兩路策略與舊設定規則見下方「通道雙路輸出設定」。H.265 供不相容瀏覽器播放時，須為對應輸出啟用相容轉碼，不能只切換 quality。

選用原生 HLS 相容：StreamGate 1.9.6 起，僅在播放端無法使用 MSE、需要原生 HLS 時，為回傳的 Gateway hls 播放 URL 加上 hls_compat=fmp4；沒有查詢參數時使用 ?，已有參數時使用 &，保留全部原始 query，尤其是播放憑據。MSE / hls.js 播放器不加此參數。它只選擇完整 fMP4 分片相容模式，不承諾 LL-HLS 低延遲，也不取代編碼相容轉碼；不新增 REST 資源，不改變播放憑據或驗簽合同。不要將它加到 playback-urls API 請求上。

StreamGate 1.9.6 起提供。可選介面：只有頁面能在使用者開啟播放器前準備按需通道時，才需要呼叫 POST /api/public/v1/channels/{id}/warm。正常播放不需要預熱；常駐通道已保持來源連線。Token 需具備獨立的 playback:warm 權限，quality 與後續播放保持一致；{"ttlMillis":15000} 表示 15 秒，上限 30 秒，回應包含 expiresAt。收到成功回應後開啟播放器；預熱失敗時仍可走正常播放流程。不要預熱全部通道或在背景持續續期。此介面只保持來源就緒，不提前生成 LL-HLS 切片，也不取代播放憑據；1.9.6 之前的版本回傳 404。

播放器恢復與來源參數變化

StreamGate 1.9.6 新增既有 WebHook/MQTT 路徑上的選用 channel.media_changed 通知，以及 GET /api/public/v1/status 的選用 channels[].media 欄位。舊版本可能省略 media，客戶端仍須接受原有 id、enabled、state 合同；沒有新增 SSE 伺服器或瀏覽器訂閱端點。通知只作補充，自建播放器仍須監聽播放器事件及畫面推進：

- WebRTC：connectionstatechange 進入 failed，或媒體軌道 ended。disconnected 可能是短時網路波動，先留出恢復時間。
- hls.js：Hls.Events.ERROR 的 fatal=true 時，依錯誤類型恢復或重建；非致命錯誤交由播放庫自動恢復。
- HTTP-FLV/WS-FLV：監聽播放庫錯誤及直播連線結束事件，釋放舊實例再重連。

播放時間持續不推進也應有界重連；暫停或頁面隱藏時勿誤判。保持單一重試任務並逐步增加間隔，關閉畫面後停止重試。[video.resize](#) 只需調整佈局，仍在播放時無需重連。相同通道可先重用原位址；更換畫質、位址或憑據到期時重新取得。重連不能替代 H.265 相容轉碼；可選 [warm](#) 不是重連通知或憑據更新介面。

選用媒體變化通知 (1.9.6)

既有 WebHook 或 MQTT 端點使用 `eventTypes:["channel.media_changed"]`，即可只訂閱媒體變化通知；StreamGate 1.9.6 的 `GET /api/public/v1/integrations/event-types` 會列出此精確類型。其他舊事件類型仍可使用並與它合併訂閱；`eventTypes:["*"]` 仍投遞全部事件，但不支援 `channel.*`。省略/留空 `eventTypes` 沿用既有預設列表，不自動加入媒體變化通知。Public 集成設定沿用 `integrations:read/integrations:write`，狀態沿用 `status:read`；仍要求有效 Professional/Enterprise 軟體授權及 `openapi` 權益，媒體投遞前再次檢查。沒有新增 `license feature` 或播放權限。

狀態欄位 (<code>channels[].media[]</code>)	說明
<code>outputStreamId</code> 、 <code>quality</code>	輸出身分及 high/fluent 畫質；只比較相同通道/輸出。
<code>current</code> 、 <code>observedAt</code>	觀測有效性與 UTC 採集時間。Core 讀取失敗或過期時保留舊值並標示 <code>current=false</code> ，不能據此認定停流。
<code>available</code>	已觀測到活動視訊描述，不代表裝置健康、已解碼畫面或全部協定就緒。
<code>mediaGeneration</code>	Gateway 不透明比較標記，不解析或排序；重啟後重新建立基線。
<code>format</code>	<code>videoCodec</code> 、 <code>width</code> 、 <code>height</code> 與 <code>origin</code> (<code>source</code> 或 <code>applied</code>)；不可用時省略。不是逐協定線路實測，不包含音訊適配。
<code>effectiveVideo</code>	選用的編碼器建立採納值： <code>codec</code> 、 <code>width</code> 、 <code>height</code> 、 <code>fps</code> 、 <code>bitrateKbps</code> 、 <code>keyframeIntervalMs</code> 。不是瞬時流量或畫質承諾。
<code>publisherGenerations</code> 、 <code>outputGeneration</code> 、 <code>selection</code>	選用的 Core 局部發布/輸出代次（十進位字串）及已選輸出策略；局部計數可能歸零，比較應使用 <code>mediaGeneration</code> 。

觀察者在既有串行媒體迴圈讀取一次 Core 聚合狀態，名義間隔兩秒而非通知延遲承諾；不啟動來源，也不逐通道探測。Public 狀態只讀快取。未觀測或舊 Core 可省略媒體欄位；超過十秒的觀測不是 `current`。每個已設定通道最多兩檔快取；刪除通道時移除快取，不保證最後通知。

去重忽略瞬時流量、封包/影格/佇列計數及尚未採納的請求設定。編碼/尺寸、有效編碼設定/實例、來源重啟身分及發布/輸出代次改變可產生新標記。主碼流與子碼流分開比較，只改子碼流輸出不改主碼流基線；可觀測的停流/恢復及同格式重啟會通知。首次為 `reason=observed`，後續為 `changed`、`stopped`、`resumed`，客戶端應容許新增值；初始化中或陳舊值不虛構新格式。

Core 目前有效的 `videoEncoding.instanceId` 與 `encoderGeneration` 參與內部比較；因此來源估計導致新碼率被採納時，即使編碼/尺寸不變也可通知。同一聚合快照的變化每檔輸出合併為一個事件；分輪觀測到的變化可能產生不同事件。

沿用既有 `envelope` 的 `eventId`、`type`、`occurredAt`、`deploymentId`、`nodeId`、`mediaResourceId`、`source`、`action`、`severity`、`message` 與字串值 `payload`。範例：

```
{ "eventId": "evt-example", "type": "channel.media_changed", "occurredAt": "2026-10-01T08:00:00Z", "mediaResourceId": "ch-001", "source": "channel", "action": "media_changed", "payload": { "outputStreamId": "ch-001__fluent", "quality": "fluent", "mediaGeneration": "media-example-2", "previousMediaGeneration": "media-example-1", "available": "true", "reason": "changed", "observedAt": "2026-10-01T08:00:00Z", "videoCodec": "h264", "width": "640", "height": "360", "formatOrigin": "applied", "bitrateKbps": "400", "publisherGenerations": { "hls-rtsp": "2" } }
```

WebHook 沿用 HMAC 標頭、重試與投遞紀錄；MQTT 沿用主題與 QoS。這是盡力投遞的取樣通知，不是持久變更日誌：兩次觀測間的短暫變化、佇列滿、接收失敗及程序重啟可能漏發，重試可能重複。以 `eventId` 去重並比較通道/輸出標記；選用低頻狀態查詢可對齊最新 `current` 狀態，不能回補中間變化。停流通知不是呼叫 `warm` 或重啟來源的命令；變更/恢復時更新播放器資訊，只有播放需要時才啟動單一有界恢復任務，不因碼率設定或尺寸改變就重連仍在推進的播放器。

新發布代次可能結束既有連線（例如 RTSP EOF），需要重連；通知不證明解碼無損，不能把非 EOF 解碼損壞當作預期的代次切換而忽略。

7. 裝置與通道管理

以下介面適用於 StreamGate 1.9.1 及以上 Professional / Enterprise。寫入請求可以提交裝置憑據；回應會移除密碼、密碼引用、金鑰、URL 使用者資訊與敏感查詢參數。

方法與路徑	Scope	用途
POST /devices	devices:write	建立裝置。
GET /devices/{id}	devices:read	查詢已脫敏的裝置設定。
PUT /devices/{id}	devices:write	更新裝置。
DELETE /devices/{id}	devices:write	刪除裝置。
POST /devices/batch	devices:write	批次 <code>enable</code> 、 <code>disable</code> 或 <code>delete</code> 。
POST /devices/onvif/discover	devices:write	探索 ONVIF 裝置。
POST /devices/onvif/resolve-stream	devices:write	解析 ONVIF 媒體串流。
POST /channels	channels:write	建立通道。
GET /channels/{id}	channels:read	查詢通道設定。
PUT /channels/{id}	channels:write	更新通道。
DELETE /channels/{id}	channels:write	刪除通道。
POST /channels/batch	channels:write	批次 <code>enable</code> 、 <code>disable</code> 、 <code>ondemand</code> 、 <code>always_on</code> 或 <code>delete</code> 。
POST /channels/{id}/ptz	ptz:write	執行 PTZ 控制。

通道雙路輸出設定

通道建立與更新使用 `channels:write`。根級 `output` 為主碼流策略；選用的 `viewing.fluent.output` 為子碼流獨立策略。兩路均不修改裝置編碼，舊設定缺少欄位時依下表保留原語義。

欄位	含義
<code>viewing.defaultQuality</code>	僅相容未傳 <code>quality</code> 的 Public API 播放位址請求，不用於控制台預覽/詳情或大螢幕。沿用 <code>auto/high/fluent</code> ； <code>high</code> 為主碼流， <code>fluent</code> 為子碼流；儲存 <code>auto/fluent</code> 仍要求已設定子來源，明確 <code>query quality</code> 覆寫它。通道 UI 已移除此選擇。
<code>viewing.fluent.source</code>	<code>profile</code> ：裝置子碼流 Profile，另傳 <code>profileToken</code> ； <code>url</code> ：裝置子碼流位址，另傳 <code>sourceUrl</code> ； <code>output</code> ：由主碼流位址的原始輸入轉碼，不傳上述來源欄位。

欄位	含義
output.mode、viewing.fluent.output.mode	空值/省略：跟隨系統；source：原始串流；compatible：相容轉碼；custom：始終轉碼。主碼流與裝置子碼流均可選四種，產生的子碼流只允許 compatible/custom；介面名稱不改 custom 值。
各路 video	compatible/custom 可設 bitrateKbps、qualityPreset、automaticPolicy、automaticBitrate、hevcBitrateReservePercent、maxWidth/maxHeight、maxFps、keyframeIntervalMs（毫秒）與 encoderPreference（舊欄位保留，實際使用全域後端）。compatible 僅將 H.265 轉 H.264，H.264 直出時不使用這些參數；source/跟隨系統不傳視訊覆寫參數。
video.automaticPolicy	選用的 economy/balanced/quality 物件，各含 sourcePercent、referenceBitrateKbps、referenceCeilingKbps；有值的新策略優先舊強度。預設值、校驗及整個物件繼承見下文。
video.hevcBitrateReservePercent	選用整數 0..200，僅用於 H.265 轉 H.264 的自動位元率。省略繼承系統，明確的 0 覆寫為不預留；獨立於 automaticPolicy，系統缺省為 35。
video.automaticBitrate	舊相容物件，含 economyPercent、balancedPercent、qualityPercent，各為整數 0 或 50..150。百分比調整內建自動目標，不是固定 kbps 或畫質分數；整個物件的繼承規則見下文。

「從主碼流位址轉碼」與主碼流共用原始輸入並獨立編碼，不串級再次編碼。編輯頁新建此子碼流時預設 custom，保持來源尺寸與影格率、檔位跟隨系統；下列通道請求片段明確設定為 640 x 360、15 fps、economy，更新時應保留其他欄位。尺寸為等比縮小上限，影格率為上限，不放大或補幀；變更子策略不覆寫主碼流策略。

雙路設定範例

```
{
  "output": {
    "mode": "compatible", "video": { "qualityPreset": "balanced", "hevcBitrateReservePercent": 0 },
    "viewing": {
      "defaultQuality": "auto",
      "fluent": {
        "source": "output",
        "output": {
          "mode": "custom",
          "video": { "qualityPreset": "economy", "maxWidth": 640, "maxHeight": 360, "maxFps": 15 }
        }
      }
    }
  }
}
```

此例主碼流明確關閉 H.265 位元率預留；子碼流省略該欄位而繼承系統。不改變兩路輸出模式或播放選擇。

設定	主碼流策略	子碼流策略
profile/url, 缺少 fluent.output	根 output	沿用根 output
profile/url, 明確 fluent.output={}	根 output	獨立跟隨系統
output 來源, 缺少 fluent.output	跟隨系統	根 output（舊特殊規則）
output 來源, 明確 fluent.output	根 output	fluent.output，須為 compatible/custom

缺少欄位不等於空物件。讀取不遷移舊設定；前端首次編輯儲存才明確轉換，並保留此前兩路語義：裝置子碼流將原根策略儲存為子策略；舊產生子碼流將原根策略移到子策略，主碼流策略儲存為跟隨系統。舊客戶端欄位與播放位址仍可使用。

自動位元率與按需拉流

兩路轉碼的 `video.qualityPreset` 可為 ""、"economy"（節省頻寬）、"balanced"（均衡）、"quality"（優先畫質）。省略或空值跟隨系統 `media.videoQualityPreset`（預設均衡），各路明確檔位覆寫系統；正數手動 `video.bitrateKbps` 優先並忽略自動策略與 H.265 預留。

在「系統設定 > 媒體與播放」的「自動轉碼」選擇節省頻寬、均衡或優先畫質。旁邊齒輪開啟「轉碼設定」，分別調整三檔畫質強度、可選位元率上限及共用關鍵影格間隔；全域編碼後端在下方「播放與資源進階設定」選擇。

自動檔位不改變來源解析度或影格率。需要縮小畫面或降低影格率時，個別設定通道主/子輸出的寬度、高度與影格率。主/子選擇「跟隨系統」時分別繼承全域轉碼模式及檔位，不會自動建立未設定的子碼流。

檔位	預設畫質強度	x264 CRF	Intel QSV QVBR 品質值
節省頻寬 economy	4	28	28
均衡 balanced	6	25	25
優先畫質 quality	8	22	22

畫質強度越高保留的細節越多，通常需要更多位元率。強度不是編碼器原生值：x264 使用 CRF+VBV；QSV 在支援時使用 QVBR；NVENC/AMF 使用其支援的品質優先 VBR。OpenH264 等缺少相應品質模式的後端使用位元率預算降級，不能視為與 CRF 等價。不同後端的數值不代表相同畫質，也不保證恢復來源已損失的細節。

三檔儲存在 `media.videoTranscodeParameters.automaticPolicy` 的 `economy/balanced/quality` 中。每檔 `qualityLevel` 為整數 1..10，0/省略使用預設值；三檔須依次不遞減。可選 `maxBitrateKbps` 為 0 或整數 256..64000 kbps，0/留空由系統估算預算；設定上限不是固定位元率，也不是包含音訊、封裝及重傳的線路硬上限。不支援品質模式的後端在實際輸出顯示 `bitrate-fallback`。

通道各路 `video.automaticPolicy` 省略、為空或全零時繼承整組系統策略，有值時整組覆寫，缺項使用內建值。通道正數 `bitrateKbps` 使用手動位元率並忽略自動檔位。全域編碼後端對所有轉碼輸出生效，舊通道 `encoderPreference` 仍可讀取但不覆寫系統選擇；通道關鍵影格間隔為 0 時使用全域值，全域預設 1000 ms。

設備 H.264 碼流在「相容轉碼」下保持直出，不使用編碼參數；H.265 才轉為 H.264。「從主碼流位址轉碼」本身就是產生獨立編碼輸出，即使主碼流為 H.264 也會編碼。兩種來源的處理規則不同，不會再次轉碼主流的編碼結果。

相容保留舊 `sourcePercent/referenceBitrateKbps/referenceCeilingKbps`、`automaticBitrate` 及 `hevcBitrateReservePercent`，僅參與內部預算/降級，不是畫質強度。預算依來源實測位元率、實際輸出尺寸與影格率計算；未測得時使用規格估計。自動策略內的舊 `maxWidth/maxHeight/maxFps` 不再改變畫面，只有通道 `video` 的明確尺寸與影格率生效。

取消不改頁面，套用只改草稿，儲存系統設定後生效。無修改套用不產生未儲存狀態。只調整共用編碼參數保留舊固定位元率，編輯自動策略或選擇「改用自動位元率」才切回自動。實際採用後端、率控模式與品質值以執行狀態為準，不新增 Public API 系統設定介面。

舊 `automaticBitrate` 保留相容，僅在繼承解析後沒有有值的新策略時使用。各分量 0 或省略為內建 100%，明確的 100 保留為覆寫。通道整個物件未提供、為空或全零時繼承整個系統強度物件；非零物件整個覆寫，缺省/零分量取內建 100%，不逐欄位繼承。例如 `{"balancedPercent":100}` 可覆寫舊系統的 90 或 120，其餘檔位為內建 100%。調整後目標須保持節省頻寬、均衡、優先畫質順序，反轉組合會被拒絕；該百分比不是固定 kbps 或畫質分數，不再作為目前系統齒輪 UI 的參數。

相容模式僅在 H.265 轉碼時使用參數；H.264 直出與原始串流不使用自動策略，也不因檔位而轉碼。儲存全域檔位或策略可能使實際採用變更值的活動輸出短暫重連，不覆寫通道明確策略或手動位元率；未使用檔位的變更不重

建目前輸出。參數不保證線路頻寬或原畫質，不新增 scope 或授權權益；儲存成功不代表編碼器已採用，可用時應讀取 current 媒體狀態。

建立裝置範例

```
POST /api/public/v1/devices
Authorization: Bearer <API_TOKEN>
Content-Type: application/json

{
  "name": "東門攝影機",
  "vendor": "generic",
  "protocol": "onvif",
  "address": "192.0.2.20",
  "username": "operator",
  "passwordRef": "<DEVICE_PASSWORD>",
  "enabled": true
}
```

onDemand=true 或批次 {"action":"ondemand","ids":["ch-001"]} 對應「按需拉流已開啟」：存在預覽、分發、錄影、分析或 GB28181 使用者時才取流，閒置後可停止；onDemand=false / always_on 對應「按需拉流已關閉」：通道啟用後保持輸入連線。API 欄位與值不變，執行中的任務仍可能保持來源連線。

8. 錄影管理

方法與路徑	Scope	用途
GET /recording/tasks	recording:read	查詢錄影任務。
POST /recording/tasks	recording:write	建立錄影任務。
GET /recording/tasks/{id}	recording:read	查詢單一錄影任務。
PUT /recording/tasks/{id}	recording:write	更新錄影任務。
DELETE /recording/tasks/{id}	recording:write	刪除錄影任務。
GET /recording/policy	recording:read	查詢預設錄影策略。
PUT /recording/policy	recording:write	更新預設策略。
POST /recording/policy/apply	recording:write	套用到指定或全部通道。

1.9.6 可按兩級選擇錄影目標：先選主路或子碼流，再選該路原始資料或處理後資料。

欄位	值	選擇
streamRole	default / secondary	主路 / 子碼流
mediaSource	source / output	所選路的原始資料 / 處理後資料

設備獨立子碼流支援兩種資料來源；從主碼流位址轉碼產生的子碼流只支援 secondary/output。未設定子碼流時不可選擇 secondary，無效組合回傳 400，不會自動改選主路。切換目標不改變錄影開關、排程或保留設定，既有片段的來源不隨目前設定改寫。

舊任務省略 streamRole 時保留主路選擇；mediaSource 省略或為空時保留原有處理策略。修改舊任務時應保留查詢回傳的未修改欄位。

錄影目錄由 StreamGate 伺服器解析與驗證，不能填寫第三方呼叫端電腦的本機路徑。設定錄影根目錄後，任務目錄必須位於該根目錄內。

9. GB28181 管理

每個資源組均支援集合 GET/POST 與單項 GET/PUT/DELETE。讀取需要 `gb28181:read`，寫入需要 `gb28181:write`。

資源	用途
<code>/gb28181/platforms</code>	設定本機 GB28181 服務平台。
<code>/gb28181/exposure-bindings</code>	將 StreamGate 通道發布到 GB28181 目錄。
<code>/gb28181/upstreams</code>	向上級平台註冊與級聯。

完整請求結構以 OpenAPI 合約為準。平台或上聯密碼可在寫入時提交，但回應不會回傳 `passwordRef`。

1.9.6 可在向上級聯設定中統一選擇供流方式；所有共享通道使用同一策略，不需要逐通道設定。

供流方式	欄位	上級點播
單碼流：主路	<code>mediaStreamMode=single、singleStreamRole=default</code>	提供通道主路輸出
單碼流：子碼流	<code>mediaStreamMode=single、singleStreamRole=secondary</code>	提供已設定的子碼流輸出
主/子雙碼流	<code>mediaStreamMode=dual</code>	未帶碼流編號或編號 0 取主路，編號 1 取子碼流

單碼流模式下，合法的主/子請求均提供所選一路。未設定子碼流的通道在上述策略下提供主路，不臨時產生子碼流；其他編號會被拒絕。舊設定未設定這兩個欄位時保留原有處理。此設定只控制即時級聯，不改變錄影選源或轉碼參數。

10. WebHook、MQTT 與執行事件

以下是 REST API 管理介面；事件依設定的 WebHook 接收位址或 MQTT broker/topic 投遞，方向與驗證方式見文件開頭。

方法與路徑	Scope	用途
GET <code>/integrations/event-types</code>	<code>integrations:read</code>	查詢可訂閱事件類型。
GET <code>/integrations/webhooks</code>	<code>integrations:read</code>	查詢 WebHook。
POST <code>/integrations/webhooks</code>	<code>integrations:write</code>	建立或更新 WebHook。
DELETE <code>/integrations/webhooks/{id}</code>	<code>integrations:write</code>	刪除 WebHook。
POST <code>/integrations/webhooks/{id}/test</code>	<code>integrations:write</code>	傳送真實測試事件。
GET <code>/integrations/mqtt</code>	<code>integrations:read</code>	查詢 MQTT 設定。
PUT <code>/integrations/mqtt</code>	<code>integrations:write</code>	更新 MQTT 設定。
POST <code>/integrations/mqtt/test</code>	<code>integrations:write</code>	傳送真實測試訊息。
GET <code>/integrations/deliveries</code>	<code>integrations:read</code>	按端點、類型與狀態查詢投遞記錄。
DELETE <code>/integrations/deliveries</code>	<code>integrations:write</code>	清除投遞記錄。
GET <code>/events</code>	<code>exports:read</code>	查詢已脫敏的執行事件。

```
{
  "name": "營運告警接收端",
  "enabled": true,
  "url": "https://ops.example.com/streamgate/events",
  "eventTypes": ["alarm.*", "recording.error", "channel.stream_error"],
  "hmacSecret": "<SHARED_SIGNING_SECRET>",
  "timeoutSeconds": 5,
  "maxAttempts": 3
}
```

hmacSecret 與 **MQTT password** 只用於寫入，查詢與儲存回應不會回傳明文。測試端點會執行真實網路投遞，聯調時應使用專用接收端。

11. 錯誤處理

錯誤回應使用 JSON。呼叫方應同時判斷 HTTP 狀態碼與穩定的結構化欄位，不要依賴完整錯誤文字進行業務分支。

```
{
  "error": "api token scope denied",
  "scope": "alarms:write"
}
```

狀態碼	含義	處理建議
400	參數或 JSON 無效。	修正類型、範圍與格式，不原樣重試。
401	Token 缺失、無效、停用或撤銷。	檢查 Authorization 標頭，必要時重建 Token。
403	授權權益或 Token scope 不足。	檢查授權頁與 Token 權限。
404	端點或資源不存在。	核對版本、資源 ID 與 URL 編碼。
405	HTTP 方法不允許。	核對 GET、POST、PUT、DELETE。
500	伺服器處理失敗。	記錄請求時間與回應摘要，檢查 StreamGate 記錄。

GET 遇到暫時性網路錯誤或 5xx 時，可採有上限的指數退避，例如 1、2、4 秒。不要無條件重試 400、401、403、404。狀態變更請求逾時後，先查詢目前狀態再決定是否重試。

12. 安全與部署

- 透過可信網路存取 StreamGate；跨網路時使用受控 HTTPS 反向代理或安全通道。
- 不同第三方系統分別建立 Token，只授予實際需要的 scope。
- Token 只儲存在伺服器端的祕密設定，不提交到程式碼儲存庫。
- 記錄中只保留 Token 前綴，不記錄完整 Authorization 標頭。
- 錄影路徑、告警擴充欄位、裝置身分與播放位址均應按部署敏感資料管理。
- 管理介面與媒體存取權限彼此獨立；啟用播放驗證時，Public API Token 不授予播放權限。
- 定期撤銷不用的 Token，並檢查 WebHook/MQTT 投遞目的地。

13. 聯調檢查表

- 1 第三方伺服器可以存取 StreamGate Web 位址與連接埠。
- 2 授權頁顯示 Professional、Enterprise 或所需的有效 API 權益。
- 3 完整 Token 已儲存到第三方系統的受保護設定。
- 4 Token 只具備目標介面需要的最小 scope。
- 5 [GET /channels](#) 回傳 200。
- 6 不在 1.8.2 實例上假設存在 1.9.1 新增端點。
- 7 客戶端明確處理逾時、[401](#)、[403](#)、[404](#) 與 [5xx](#)。
- 8 記錄、截圖與匯出內容不含完整 Token 或裝置密碼。
- 9 WebHook/MQTT 測試使用專用接收端，生產啟用前已完成檢查。
- 10 播放已按實際部署的播放驗證策略完成測試。

14. 配套文件

具備有效 Professional / Enterprise 授權時，[GET /api/public/v1/openapi.json](#) 可取得執行中實例內建的機器可讀合約，用於版本探測與工具匯入。Professional / Enterprise 套件另包含 [docs/streamgate-public-api-v1.openapi.yaml](#)，可匯入 Swagger Editor、Postman 或相容的程式碼產生工具。

Postman 集合只在客戶明確需要聯調時另行提供。HTML 是文件維護資產，不進入執行套件。

本文件中的主機名稱、資源識別碼、Token、憑據與業務資料均為範例，請替換為實際部署值。